



# International Journal of Data Science and Big Data Analytics

Publisher's Home Page: <https://www.svedbergopen.com/>



Review Article

Open Access

## Algorithms in future insurance markets

Malgorzata Śmietanka<sup>1\*</sup>, Adriano Koshiyama<sup>2</sup> and Philip Treleaven<sup>3</sup>

<sup>1</sup>University College London, Gower St, Bloomsbury, London WC1E 6BT, United Kingdom. E-mail: malgorzata.wasiewicz.17@ucl.ac.uk

<sup>2</sup>University College London, Gower St, Bloomsbury, London WC1E 6BT, United Kingdom. E-mail: adriano.koshiyama.15@ucl.ac.uk

<sup>3</sup>University College London, Gower St, Bloomsbury, London WC1E 6BT, United Kingdom. E-mail: p.treleaven@ucl.ac.uk

### Article Info

Volume 1, Issue 1, February 2021

Received : 09 November 2020

Accepted : 12 January 2021

Published : 05 February 2021

doi: [10.51483/IJDSBDA.1.1.2021.1-19](https://doi.org/10.51483/IJDSBDA.1.1.2021.1-19)

### Abstract

This paper reviews the impact of data science and artificial intelligence (AI) on future ‘data-driven’ insurance markets. The impact of insurance automation (driven by so-called Black Swan<sup>1</sup> events such as Covid-19) mirrors the impact of algorithmic trading that changed radically the capital markets (Koshiyama et al., 2020). The data science technologies driving change include: Big data, AI analytics, Internet of Things, and Blockchain technologies. These technologies are important since they underpin the automation of the insurance markets and risk analysis, and provide the context for the algorithms, such as AI machine learning and computational statistics, which provide powerful analytics capabilities. New AI algorithms are constantly emerging, with each ‘strain’ mimicking a new form of human learning, reasoning, knowledge, and decision-making. The current main disrupting forms of learning include deep learning, adversarial learning, federated learning, transfer and meta learning. Albeit these modes of learning have been in the AI/ML field more than a decade, they are now more applicable due to the availability of data, computing power and infrastructure. These forms of learning have produced new models (e.g., long short-term memory, generative adversarial networks) and leverage important applications (e.g., Natural Language Processing, Adversarial Examples, Deep Fakes, etc.). These new models and applications will drive changes in future insurance markets, so it is important to understand their computational strengths and weaknesses. The contribution of this paper is to review the data science technologies and specifically AI algorithms, their computational strengths and weaknesses, and discuss their future impact on the insurance markets.

**Keywords:** Insurance, AI, Machine Learning, Algorithms review

© 2021 International Journal of Data Science and Big Data Analytics. This is an open access article under the CC BY license (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## 1. Introduction

### 1.1. Data-driven insurance industry

Data science technologies and artificial intelligence (AI) is revolutionizing the insurance marketplace and creating a new generation of InsurTech companies that are ‘data-driven’ (cf. Amazon). In China Zhong An (a digital insurance collaboration of Alibaba, Tencent and Ping An) underwrote over 630 million insurance policies in its first year of operation (The Digital Insurer forum). So-called ‘Black Swan’ events such as Covid-19 will clearly accelerate change in how we work, and buy products and services. The insurance industry already collects huge volumes of data and thus

\* Corresponding author: Malgorzata Śmietanka, University College London, Gower St, Bloomsbury, London WC1E 6BT, United Kingdom. E-mail: malgorzata.wasiewicz.17@ucl.ac.uk

<sup>1</sup> Black Swan – a massive unpredictable event that has potentially severe consequences.

has major opportunities for utilizing machine learning to uncover hidden patterns, unknown correlations, customer preferences and risk. At one end, this allows precise risk profiling on institutions and individuals and at the other, building complex (ecology) models for long-term prediction such as natural disasters.

For the Insurance industry, the data science technologies provide, on the one hand, unprecedented volumes of data and analytics tools for analysis, but on the other ‘revolutionary’ innovations present new business challenges. To understand the opportunities offered by ‘data-driven’ insurance, it is necessary to understand the data science technologies contributing to the ‘perfect storm’.

### 1.2. How AI will transform the insurance sector

In insurance, AI-based analytics will help insurance companies to drive new business by allowing companies to personalize insurance products and better respond to existing clients’ needs. More accurate underwriting models and constant learning from new data will be reflected in premiums that are often overestimated because of lack of data. There is a number of ways to leverage machine learning techniques in the insurance industry, including:

- **Automated and Personalized product offerings.** Automated processes have an important impact on the insurance industry, to analyze large portions of data and gain a more specific perspective of the client’s activities. Insurers can offer personalized products and solutions that are based on the specific needs of narrow segments. One example of AI in customer service is the Lemonade insurance app (Lemonade.com), which makes use of AI-powered Chatbots to assist its customers. AI Chatbot provides personalized policies to customers in less than two minutes.
- **Behavioral Product Pricing.** Telematics, wearable sensors and smart watches (known as Internet of Things) provide a wealth of information for insurance companies to profile the client’s risk. AI algorithms can provide dynamically changing premiums based on health markers or driving behaviors, which reflect the risk profile of the insured. In this way, insurers can become more involved in their policyholders health or safe driving habits.
- **Improved Risk Assessment.** When compared to humans, machine learning can deliver additional predictions. Insurance companies can construct targeted predictions on coverage changes, and possible losses for policies and manage risks more effectively using different sources of data.
- **Enhanced Fraud Detection.** Fraud is one of the biggest problems for the insurance companies. FBI estimated around US\$40 billion per year lost to insurance fraud (FBI.gov). Machine learning algorithms can reduce human errors and identify unobserved fraud patterns by identifying exceptions. Fraud detection solutions analyze massive amounts of data from multiple sources. AI powered scoring system then analyses each claim as to how likely fraudulent it might be.
- **Business Processes Automation.** AI-powered systems may eliminate tedious, repetitive and mundane processes across the whole organization. AI-based products can help in analyzing complex documents and extracting vital information from them.

The AI revolution will go even further leading to so-called Computable (insurance) Contracts ([Marano and Noussia, 2019](#)). Computable contracts are legal specifications that a computer can read, understand, verify and execute; and therefore automate.

Currently, Big Data technologies are applied to predict risks and claims, to monitor and to analyze them in order to develop effective strategies for customer’s attraction and retention. In Table 1, we listed examples of use cases where machine learning techniques are applied:

| Table 1: AI use cases in insurance |  |
|------------------------------------|--|
| Insurance area                     | Use case   |
| Healthcare Insurance               | Fraud and abuse detection system ( <a href="#">Ilker et al., 2015</a> ).   |
| Motor Insurance                    | Fraud detection system – A multiple classifier system based on Random Forest, Principle Component Analysis and Potential Nearest Neighbor is proposed ( <a href="#">Yaqi et al., 2018</a> ).     |
| Non-Life Reporting                 | Multilayer Perceptron applied to the problem of the prediction of insolvency of non-life insurance companies, upon the basis of a set of financial ratios ( <a href="#">Díaz et al., 2005</a> ). |
| Claims Reserving                   | Regression trees used to calculate claims reserves on individual claims data ( <a href="#">Wüthrich, 2018</a> ).   |

### 1.3. Challenges for insurance industry under AI revolution

Challenges with adopting AI can be grouped into two areas: (a) those that stem from algorithms (e.g., explainability/interpretability, robustness, fairness), and (b) those related to the dynamic nature of AI.

Below we unpack the second group that can be further divided into: a) technology, b) data strategy, c) a culture.

- **Technology stack/ lack of technical knowledge** – to deploy and implement AI applications, organizations need the knowledge of the current AI advancements and technologies and their shortcomings. Adopted technologies need to be relevant to the business requirements as well as be flexible enough to enable integration of different data sources (both internal and external). AI solutions require high degree of computation speed that imposes further infrastructure requirements.
- **Data strategy** – Big data obtained from customers, core systems, brokers need to be appropriately managed. AI may be working with different types of data (e.g., text, voice, video, image, and sensors - IoT). To fully leverage AI in the industry all available data sources should be utilized. Data may be structured, semi-structured and unstructured. Another classification of data can be into real-time, historical, third-party data. Pulling all those data sources together, managing data quality and creating value out of that can be a really challenging task for the industry. Also, insurance companies need an efficient infrastructure for controlling their data (data shouldn't be disintermediated by brokers).
- **Aversion towards AI** – both staff and customers need to be convinced that innovation that AI is utilized to empower decision-making that will benefit everyone (improved efficiency for staff, more accurate risk pricing and personalized products for customers).

An accurate role model and illustration of future data-driven insurance is the introduction of algorithmic trading in the Capital Markets (Treleaven, et al., 2013) over the past 18 years. Notable is firstly the collection and analysis of increasingly huge volumes of real-time and historic data (e.g., financial, economic, social, alternative), and secondly the increasing use of AI algorithms that can switch dynamically to respond to changes in the market microstructure (Koshiyama et al., 2020). Insurance companies will seek increasingly to collect and control their data; and also automate to interact directly with their clients.

Next, we will unpack the new 'disruptive' technologies.

## 2. Data science technologies driving change

To provide context, our review divides the data science technologies into:

- **Data technologies** – includes solutions for data management and collection, as well as services that are based on data generated by both human and machines.
- **Algorithm technologies** – new forms of 'statistics', such as machine learning, computational statistics, and complex systems (e.g., deep neural networks, Monte Carlo simulation).
- **Analytics technologies** – covering the application of the data technologies (e.g., natural language, sentiment analysis and behavioral analysis).
- **Infrastructure technologies** – providing the infrastructure for information management and automation (e.g., Blockchain- based digital marketplace, computable contracts).

### 2.1. Data technologies

- Big data- the collection and analysis of huge volumes of historic and real-time information (e.g., financial, economic, social media, alternative);
- Cloud Computing- on-demand availability of computer system resources such as data storage or computing power.
- Chatbots – data provided by computer programs that simulates human conversation through voice commands or text chats or both; using natural language processing (NLP) and sentiment analysis to understand the conversation.
- Internet of Things (IoT) - the inter-networking of 'smart' physical devices, vehicles, buildings, etc. that enable these objects to collect and exchange data.

### 2.2. Algorithm technologies

For completeness, this section unpacks algorithms across three domains (Table 2): Computational Statistics (e.g., Monte Carlo methods), AI and ML (e.g., Artificial Neural Networks), and Complex Systems (e.g., Agent-Based systems). While there may be some debate over the terminology, we find the classification helpful to distinguish between relatively well-established methods and more cutting-edge technologies.

**Table 2: Algorithm domains**

**Computational statistics** – Computationally intensive statistical methods.

**AI algorithms** – Mimicking a new form of human learning, reasoning, knowledge, and decision-making.

- Knowledge or rule-based systems
- Evolutionary algorithms
- Machine learning

**Complex systems** – System featuring a large number of interacting components whose aggregate activity is nonlinear.

### 2.2.1. Computational statistics

Computational statistics models refers to computationally intensive statistical methods including Resampling methods (e.g., bootstrap and cross-validation), Monte Carlo methods, Kernel Density estimation and other semi and non-parametric methods, and generalized additive models (Efron and Hastie, 2016; Wood, 2017). Examples include: (a) Resampling methods – a variety of methods for doing one of the following: (i) estimating the precision of sample statistics using subsets of data (e.g., jack-knifing) or drawn randomly from a set of data points (e.g., bootstrapping); (ii) exchanging labels on data points when performing significance tests (e.g., permutation tests); (iii) validating models by using random subsets (e.g., repeated cross-validation); (b) Monte Carlo methods – a broad class of computational algorithms that rely on repeated random sampling to approximate integrals, particularly used to compute expected values including those meant for inference and estimation (e.g., Bayesian estimation, simulated method of moments); (c) Kernel Density estimation – are a set of methods used to approximate multivariate density functions from a set of data points; it is largely applied to generate smooth functions, reduce outliers effects and improve joint density estimations, sampling, and to derive non-linear fits; (d) Generalized Additive Models – a large class of nonlinear models widely used for inference and predictive modeling (e.g., time series forecasting, curve-fitting, etc.); and (e) Regularization methods – Regularization methods are increasingly used as an alternative to traditional hypothesis testing and criteria-based methods, for allowing better quality forecasts with a large number of features.

### 2.2.2. AI and machine learning

This AI continuum of epistemological models spans three main communities: (a) Knowledge-based or heuristic algorithms (e.g., rule-based) – where knowledge is explicitly represented as ontologies or IF-THEN rules rather than implicitly via code (Giarratano and Riley, 1998); (b) Evolutionary or metaheuristics algorithms – a family of algorithms for global optimization inspired by biological evolution, using population-based trial and error problem solvers with a metaheuristic or stochastic optimization character (e.g., Genetic Algorithms, Genetic Programming, etc.) (Poli et al., 2008; Brownlee, 2011); and (c) Machine learning algorithms – a type of AI program with the ability to learn without explicit programming, and can change when exposed to new data; mainly comprising Supervised (e.g., Support Vector Machines, Random Forest, etc.), Unsupervised (e.g., K-Means, Independent Component Analysis, etc.), and Reinforcement Learning (e.g., Q-Learning, Temporal Differences, Gradient Policy Search, etc.) (Hastie et al., 2009; Sutton and Barto, 2018). Russell and Norvig (2020) provide an in-depth view of different aspects of AI.

### 2.2.3. Complex systems

Lastly, a complex system is any system featuring a large number of interacting components (e.g., agents, processes, etc.) whose aggregate activity is nonlinear (not derivable from the summations of the activity of individual components) and typically exhibit hierarchical self-organization under selective pressures (Taylor, 2014; Barabási, 2016). Examples include: (a) Cellular automata – a collection of cells arranged in a grid, such that each cell changes state as a function of time according to a defined set of rules that includes the states of neighboring cells; (b) Agent-based models – a class of computational models for simulating the actions and interactions of autonomous agents (individual or collective entities such as organizations or groups) with a view to assessing their effects on the system as a whole; (c) Network-based models – a complex network is a graph (network) with non-trivial topological features-features that do not occur in simple networks such as lattices or random graphs but often occur in graphs modeling of real systems; and and (d) Multi-Agent systems – this subarea focus on formulating cooperative-competitive policies to a multitude of agents with the aim to achieve a given goal; this topic has significant overlap with Reinforcement Learning and Agent-based models.

| Table 3: Landscape of algorithms and their applications |                                 |                                 |                               |                            |
|---|---------------------------------|---------------------------------|-------------------------------|----------------------------|
|   | NLP and sentiment analysis      | Fraud detection                 | Reporting/ Actuarial modeling | Compliance/ regulation     |
| Computational Statistics                                | (Cambria <i>et al.</i> , 2013)  | (Juszczak <i>et al.</i> , 2008) | (Bäumer, 2000)                | (Yang and Koshiyama, 2019) |
| Machine Learning  | (Kolchyna <i>et al.</i> , 2015) | (Adewumi and Akinyelu, 2017)    | (Richman, 2018)               | (van Liebergen, 2017)      |
| Complex Systems   | (Batrinca and Treleaven, 2015)  | (Xu and Chen, 2005)             | (Allan <i>et al.</i> , 2012)  | (May <i>et al.</i> , 2008) |

As an illustration of this landscape of algorithms and research, Table 3 presents a non-exhaustive list of references that links each class of algorithms to applications in different areas of insurance.

### 2.3. Analytics technologies

Analytics technologies are used by many insurers to collect and predict customer behavior.

#### 2.3.1. Behavioural/Predictive Analytics

Predictive Analytics is the analysis of large and varied data sets to uncover hidden patterns, unknown correlations, and customer preferences etc. to help make informed decisions. Predictive analytics is the practice of extracting information from historical and real-time data sets to determine patterns and predict future outcomes and trends. Predictive analytics ‘forecasts’ what might happen in the future with an acceptable level of reliability, and includes what-if scenarios and risk assessment.

Insurance companies can use predictive analytics for: (a) pricing and risk selection; (b) identifying risks (e.g., fraud, cancellation); and (c) identifying outlier claims.

### 2.4. Infrastructure technologies

For the Insurance industry key infrastructure technologies driving change are: (a) Federated learning, (b) Computable insurance contracts; and (c) Blockchain-based digital marketplaces.

#### 2.4.1. Federated learning

Federated learning is an important emerging technique, given the value and sensitivity of data (e.g., financial, business, social, alternative and regulatory). With federated learning, the focus is on decentralized framework enabling multiple data holders to collaborate and to converge to a common model without exchanging raw data.

We distinguish the following categories of Federated learning:

- Horizontal Federated learning (HFL) – Assumes that datasets from different participants share the same feature space but may not share the same sample ID space (Figure 1).

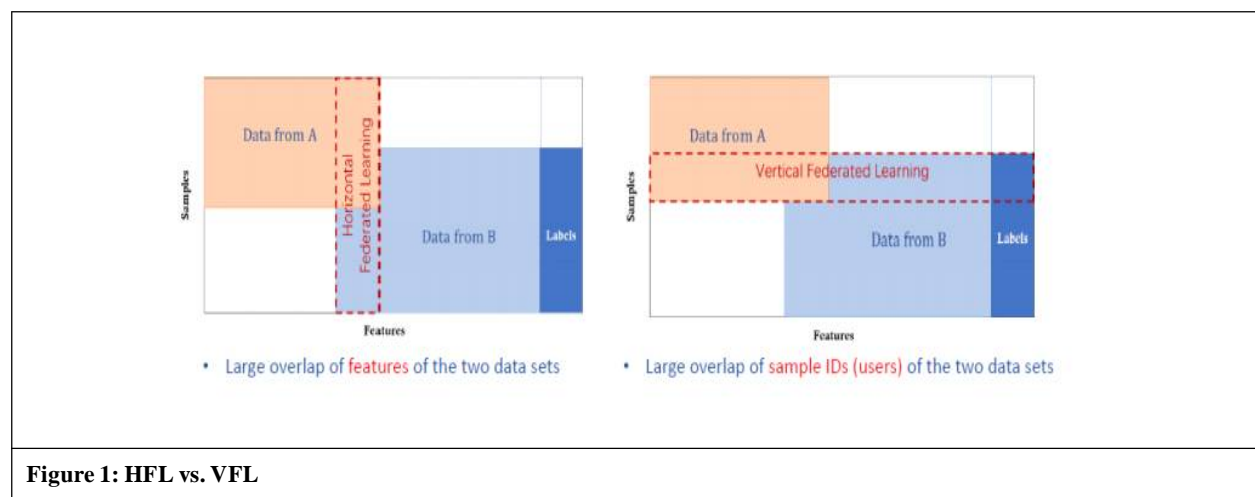


Figure 1: HFL vs. VFL



- Vertical Federated Learning (VFL) – participants share the same ID space but have different feature spaces, while label information is owned by one participant (see Figure 4). One possible use case is two e-commerce companies and a bank which collaborate on training a model to recommend personalized loans for users based on their online shopping behaviors through VFL (Yang et al., 2019). This use case can easily be adopted to the insurance domain, where NHS collaborates with medical insurance providers to provide personalized medical insurance products.

Open source FL architectures include Google's TensorFlow Federated (<https://www.tensorflow.org/federated>), PySyft (<https://github.com/OpenMined/PySyft>) and Webank's Federated AI Technology Enabler FATE (Li et al., 2019).

#### 2.4.2. Computable insurance contracts

Computer-readable and executable legal specifications are set to have a profound impact on business and legal services. So-called *Computable* contracts are legal specifications that a computer can read, understand, verify and execute, and therefore automate. The challenge is to specifying a contract that can be composed and read by professionals, and can also be translated into a domain-specific computer-readable specification such as XML ([www.service-architecture.com/articles/xml/insurance\\_xml.html](http://www.service-architecture.com/articles/xml/insurance_xml.html)). Here the big potential is automating high volume low cost domestic insurance (e.g., health, vehicle), without human intervention. Smart Contracts - are simply rules, possibly computer programs, which codify transactions and contracts with the intent that the records managed by the distributed ledger are authoritative with respect to the existence, status and evolution of the underlying legal agreements they represent. Smart contract technology has the potential to automate laws and statutes.

##### 2.4.2.1. Blockchain technologies

The core technologies are: Distributed Ledger Technology (DLT) and smart contracts. DLT is a decentralized database where transactions are kept in a shared, replicated, synchronized, distributed bookkeeping record, which is secured by cryptographic sealing. The key distinction between 'distributed ledgers' and 'distributed databases' is that nodes of the distributed ledger cannot/do not trust other nodes—and so must independently verify transactions before applying them.

##### 2.4.3. Blockchain-based digital marketplaces

With the Uberization and globalization of insurance services, a major opportunity exists to create an 'Amazon/Alibaba' for digital insurance services using Blockchain technology built on top of the Internet. Blockchain marketplace is under development in Singapore ([www.sginnovate.com/events/future-blockchain-based-data-marketplaces-challenges-and-impact](http://www.sginnovate.com/events/future-blockchain-based-data-marketplaces-challenges-and-impact)) and Dubai ([www.smartdubai.ae/](http://www.smartdubai.ae/)). However, the most comprehensive 'digital' Blockchain infrastructure program is Estonia's *e-Estonia* (<https://e-estonia.com/>) where every citizen has a digital identity, digital signature and personal record, and virtually all government services are digital and online.

In the next section we will classify and introduce machine learning algorithms.

### 3. Machine learning paradigms

The great computational strength of ML algorithms is their ability to 'learn' without explicit programming. Understanding computational 'learning' is likely to have a profound effect on future science, in both artificial and natural (biological) systems.

As illustrated by examples in Table 4, the driving forces of new ML algorithms are broadly a combination of the classical *trio* of Supervised, Unsupervised and Reinforcement Learning, with the *disruptors*: Deep Learning, Adversarial

**Table 4: Algorithms emerging by interaction between different learning paradigms**

|                         | Supervised   | Unsupervised   | Reinforcement  |
|-------------------------|--|--|--|
| Deep Learning           | Deep Convolutional Neural Networks, Deep Recurrent Neural Networks (Goodfellow et al., 2016).                                    | Deep Autoencoders (Goodfellow et al., 2016), Deep clustering (Caron et al., 2018).   | Deep Q-Learning, Trust Region Policy Optimization, Asynchronous Advantage Actor Critic (Arulkumaran et al., 2017)  |
| Adversarial Learning    | Adversarial Semi-supervised Learning (Miyato et al., 2017); Adversarial Robustness in Supervised Learning (Nicolae et al., 2019) | Adversarial Autoencoders (Makhzani et al., 2016); Adversarial Representation Learning (Chen et al., 2016), Generative Adversarial Networks (Goodfellow et al., 2014) | Adversarial Policies (Gleave et al., 2019); Robust Adversarial Reinforcement Learning (Pinto et al., 2017)         |
| Transfer/ Meta Learning | OPEN-GPT (Radford et al., n.d.); BERT (Devlin et al., 2018) ; MedicalNet (Chen et al., 2019)                                     | Bayesian Unsupervised One-Shot learning (Fei-Fei et al., 2003); Embeddings from Language Model (Siddhant et al., 2019)   | Darla (Higgins et al., 2017); Deep Transfer Reinforcement Learning for Text Summarization (Keneshloo et al., 2019) |

Learning, Transfer and Meta Learning. This interaction constantly yields new models (e.g., Long Short-Term Memory (LSTMs), Generative Adversarial Networks (GANs)) and applications (e.g., Natural Language Processing, Object Recognition, Forecasting, etc.).

### 3.1. Supervised, unsupervised, reinforcement

ML firstly subdivides into:

- **Supervised learning:** Given a set of inputs/independent variables/predictors  $x$  and outputs/dependent variables/targets  $y$ , the goal is to learn a function  $f(x)$  that approximates  $y$ . This is accomplished by supervising  $f(x)$ , that is, providing it with examples  $(x_1, y_1), \dots, (x_n, y_n)$  and feedback whenever it makes mistakes or accurate predictions.
- **Unsupervised learning:** Given several objects/samples/transactions  $x_1, \dots, x_n$ , the goal is to learn a hidden map  $h(x)$  that can uncover a hidden structure in the data. This hidden map can be used to 'compress'  $x$  (aka dimensionality reduction) or to assign to every  $x_i$  a group  $c_k$  (aka clustering or topic modeling).
- **Reinforcement learning:** Given an environment formed by several states  $s_1, s_2, \dots, s_n$  an agent, and a reward function, the goal is to learn a policy  $\pi$  that will guide an agent actions  $a_1, a_2, \dots, a_k$  through the state space so as to maximize occasional rewards.

Figure 2 provides an illustration of these key learning paradigms. Suppose a database of financial reports is available. If some of them have been historically labeled as positive and negative, we can leverage this to automatically tag future documents. This can be accomplished by training a Learner in a supervised fashion. If these documents were unstructured, and spotting relations or topics is the goal (political events, economic data, etc.), a Learner trained in an unsupervised manner can help uncover these hidden structures. Also, these documents can characterize the current state of the insurance markets. Using that, a Learner can decide which actions should be taken in order to maximize profits, detect new risks, etc. By interacting and gaining feedback from the environment (Markets), the Learner can reinforce some behaviors so to avoid future losses or inaccurate decisions.

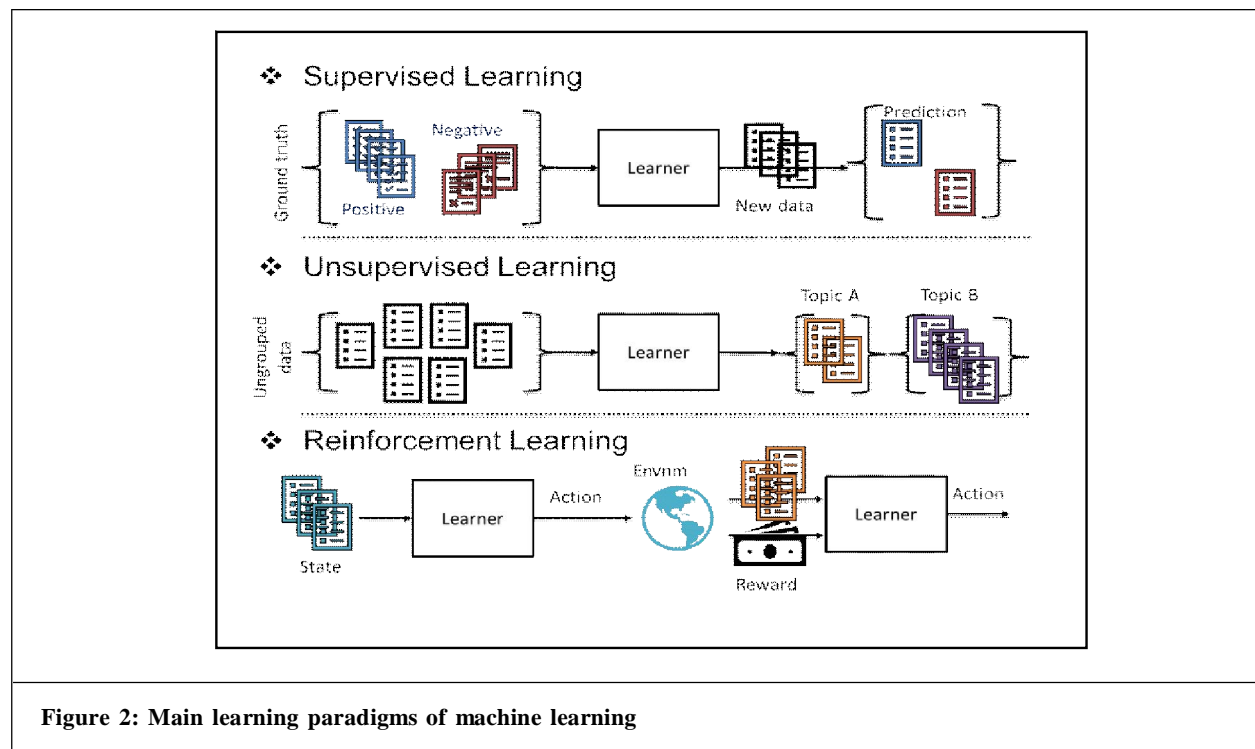


Figure 2: Main learning paradigms of machine learning

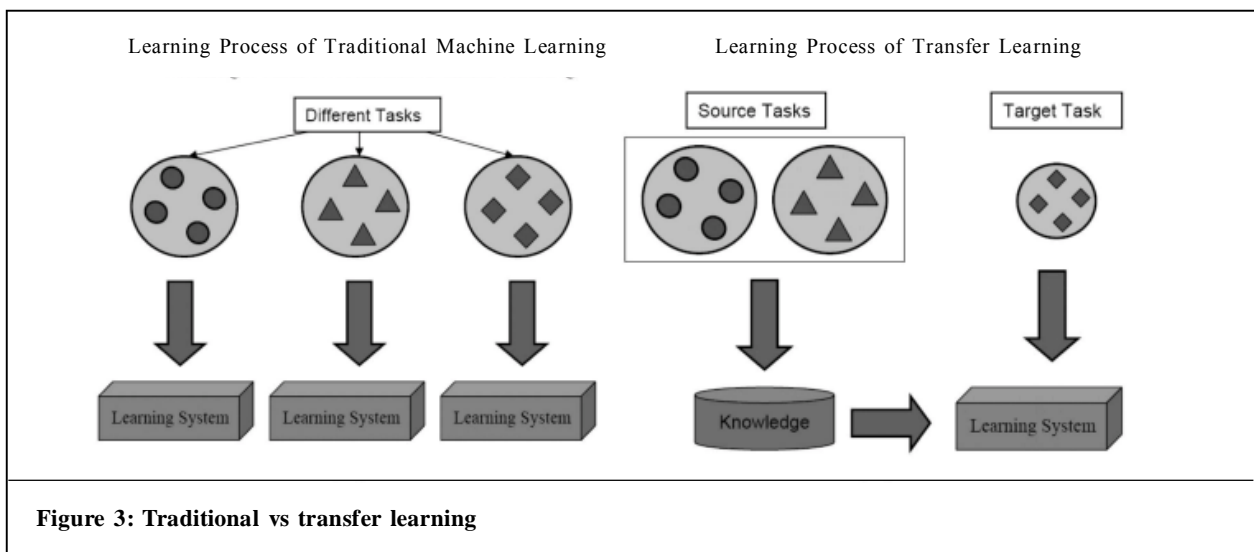
### 3.2. Deep learning, adversarial learning, transfer/meta learning

These new forms of learning are 'disrupting' the current models prevalent in Supervised, Unsupervised and Reinforcement learning. They are not only powering new solutions and applications (e.g., driverless vehicles, smart-speakers, etc.) but they are making the resolution of previous problems cheaper, faster and more scalable. The second subdivision is:

- **Deep Learning** – Deep learning algorithms attempt to model high-level abstractions in data by using multiple processing layers, with complex structures or otherwise, composed of multiple non-linear transformations. Hence,

the mapping function we are attempting to learn can be broken down into several compositional operations  $f(x) = f_1 \cdot f_2 \cdot f_3 \cdot \dots \cdot f_n(x)$ . Various deep learning architectures such as deep neural networks, convolutional deep neural networks, deep belief networks and recurrent neural networks have been applied to fields like computer vision, automatic speech recognition, natural language processing, audio recognition and bioinformatics where they have been shown to produce state-of-the-art results on various tasks (Goodfellow et al., 2016; Chollet, 2017).

- **Adversarial learning** – Adversarial machine learning is a technique employed in the field of machine learning which attempts to ‘fool’ models through malicious input. More formally, assume a given input  $x$  associated to a label  $c$  and a machine learning model  $f$  such that  $f(x) = c$ , that is,  $f$  can perfectly classify  $x$ . We consider  $x^*$  an adversarial example if  $x^*$  is indistinguishable from  $x$  and  $f(x^*) \neq c$ . Since they are automatically crafted, these adversarial examples tend to be misclassified more often than is true of examples which are perturbed by noise (Szegedy et al., 2013; Kurakin et al., 2016). Adversarial examples can be introduced during the training of models, making them more robust to attacks



from adversarial agents. Typical applications involve increasing robustness in neural networks, spam filtering, information security applications, etc. (Huang et al., 2011).

- **Transfer Learning** – Focuses on utilizing knowledge gained while solving one problem to solve related ones. A closely related technique is multitask learning framework which aims in learning multiple tasks at the same time, even when they are different. A common approach is to uncover the common (latent) features that can benefit each individual task. Figure 3 depicts the difference between traditional approach of building and training machine learning models and the methodology following transfer learning principles. Transfer learning is broadly used in classification, regression and clustering problems. Raina et al., (2006) and Dai et al., (2007) proposed transfer learning to learn text data across domains.

Next we look at how ‘disruptive’ models will change the Insurance Industry.

#### 4. Algorithms in future insurance markets

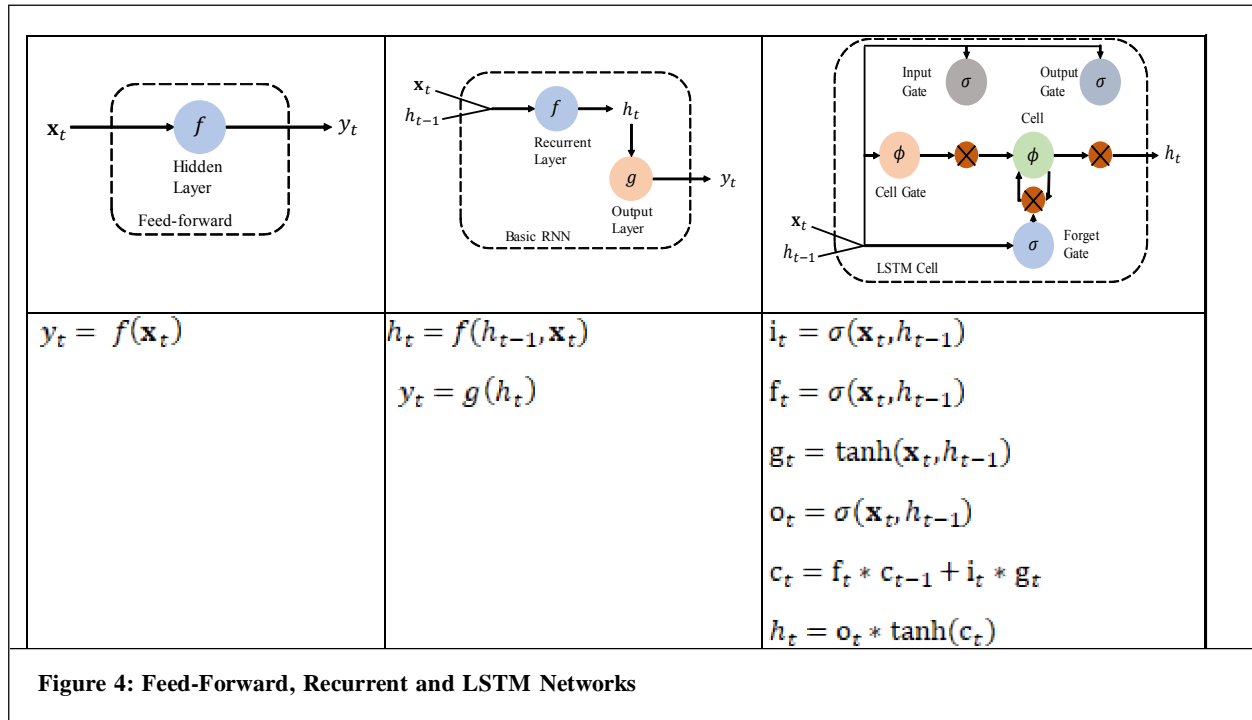
New AI algorithms are constantly emerging; examples include LSTMs – a type of deep recurrent neural network capable of learning arbitrary long-term dependencies; GANs – an architecture comprised of two networks, pitting one against the other (thus the ‘adversarial’); and Transfer and Meta learning families – paradigms to reuse experience gained by solving predecessor problems as well as fine-tuning them for unseen tasks.

This section introduces LSTMs and GANs, their typical applications and potential uses across insurance markets.

##### 4.1. Long Short-Term Memory (LSTM)

LSTM networks (Hochreiter and Schmidhuber, 2006) are a type of recurrent neural network (RNN) capable of learning order dependence in sequence prediction problems, by keeping information about past inputs for an amount of time that is not fixed a priori, but rather depends on its weights, number of stacked layers and on the input data. Whereas a simple feed-forward neural network treats its inputs as independent, an RNN uses previous input sources within the calculations to recognize a data’s sequential characteristics. Figure 4 illustrates the difference between a typical feed-forward, a recurrent neural and a LSTM network.





More formally, we can express their distinctions by their mathematical steps to produce an output:

- **Feed-forward NN**

- $y_t = f(x_t)$ ,

with  $x_t$ ,  $y_t$  as input and output at time  $t$  respectively, and  $f(x) = f_1 f_2 f_3 \dots f_n(x)$  as similar as a deep neural network computation.

- **Basic RNN**

- $h_t = f(h_{t-1}, x_t)$

- $y_t = g(h_t)$

with  $g$  representing a mapping from the hidden state  $h_t$  back to the ‘visible’ output state  $y_t$  at time  $t$ ; this state can be broadly understood as a compressed representation of the sequence being historically observed so far.

- **LSTM**

- $i_t = \sigma(x_t, h_{t-1})$

- $f_t = \sigma(x_t, h_{t-1})$

- $g_t = \tanh(x_t, h_{t-1})$

- $o_t = \sigma(x_t, h_{t-1})$

- $c_t = f_t * c_{t-1} + i_t * g_t$

- $h_t = o_t * \tanh(c_t)$

- $y_t = g(h_t)$

with  $i_t, f_t, g_t, o_t$  denoting the input, forget, cell and output gates, respectively;  $\sigma$  the sigmoid function,  $\tanh$  the hyperbolic tangent function, and  $*$  the Hadamard product. The input, forget, and output gates are responsible for the transfer of information across the architecture, whilst the cell  $c_t$  accumulates the information processed across these gates. As their name imply, the input gate decides how much the time  $t$  input and previous hidden state still matters for the current moment; the forget gate acts as a ‘reset’, zeroing the accumulated information stored in the cell; the output gate modulates what part of the current cell state make it to the final hidden state.

Overall, basic RNNs are a network of neuron-like nodes organized into successive ‘layers.’ Each node in a given layer connected with a directed (one-way) connection to every other node in the next successive layer. In summary, sequential information (e.g., financial time series) is preserved in the recurrent network’s hidden state, which manages to span many time steps as it cascades forward to affect the processing of each new example. LSTMs are designed to

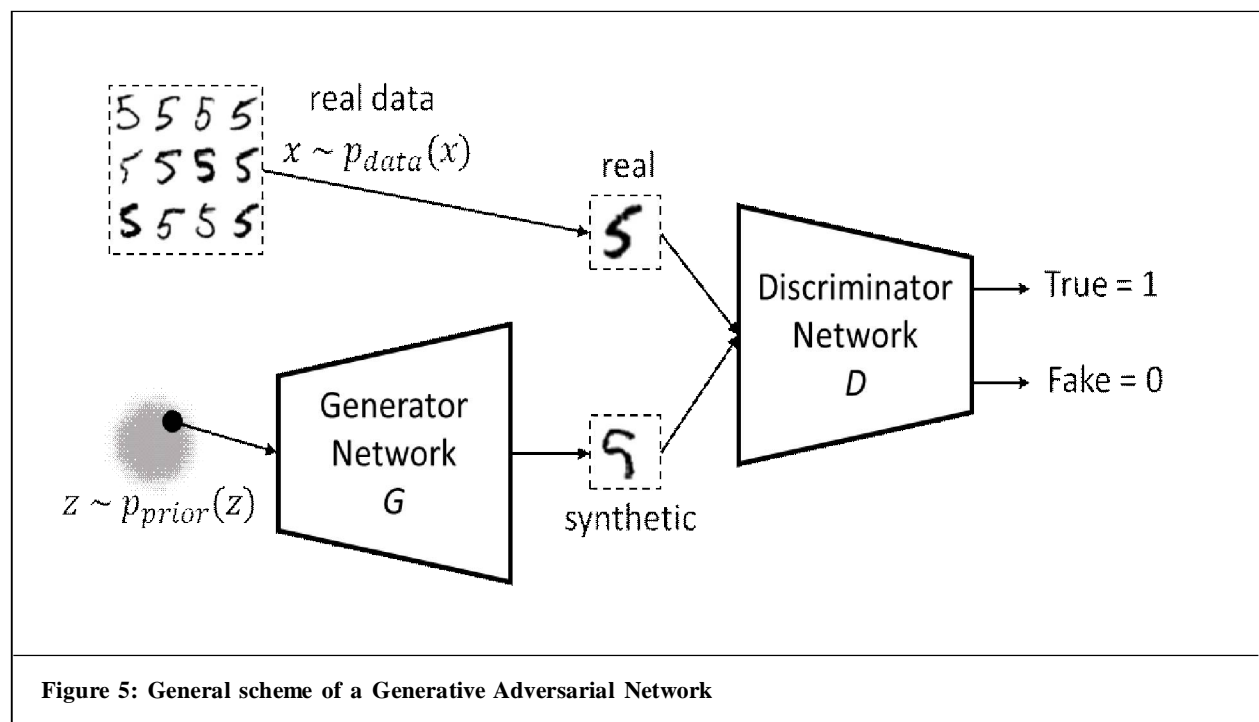
overcome one of the drawbacks to basic RNNs, called the vanishing gradient problem (Goodfellow et al., 2016), in which performance of the neural network suffers because it can't be trained properly. LSTM units categorize data into short-term and long-term memory cells. This enables identification of which data is important, should be remembered, and looped back into the network, and what data can be forgotten.

**Traditional use cases:** The LSTM model has been found to be highly successful in many applications, such as unconstrained handwriting recognition (Graves et al., 2009), speech recognition (Graves et al., 2013) (Graves and Jaitly, 2014), handwriting generation (Graves, 2014), machine translation (Sutskever et al., 2014), image captioning (Kiros et al., 2014) (Xu et al., 2016), and parsing (Vinyals et al., 2015).

**Applications in insurance markets:** Weishan et al. (2016) employed CNN (Convolutional Neural Network using 1-D Convolution and Pooling Convolutional neural network) for learning driving styles from the GPS data. Further, (Saleh et al., 2017) utilized LSTM recurrent neural networks for classification of driving behavior based on sensor data from smartphones. (Diao and Wang, 2019) has explored a way to base insurance premium income on LSTM.

#### 4.2. Generative Adversarial Networks (GANs)

GANs (Goodfellow et al., 2014) is a modeling strategy that employs two Neural Networks: a Generator ( $G$ ) and a Discriminator ( $D$ )—Figure 5. The  $G$  is responsible for producing produce a rich, high dimensional vector attempting to replicate a given data generation process; the  $D$  acts to separate the input created by the  $G$  and of the real/observed data generation process. They are trained jointly, with  $G$  benefiting from  $D$ 's incapability to recognise true from generated data, whilst  $D$  loss is minimized when it is able to classify correctly inputs coming from  $G$  as fake and the dataset as true. Competition drive both networks to improve their performance until the genuine data is indistinguishable from the generated one.



**Traditional use cases:** Overall, GANs have been successfully applied to image and text generation (Creswell et al., 2017); BigGAN is a very successful example of using GANs to create high fidelity natural image synthesis and representation learning (Brock et al., 2018).

**Applications in insurance markets:** Kuo (2019) utilized CTGAN architecture proposed by Xu et al. based on GAN for synthesizing insurance datasets.

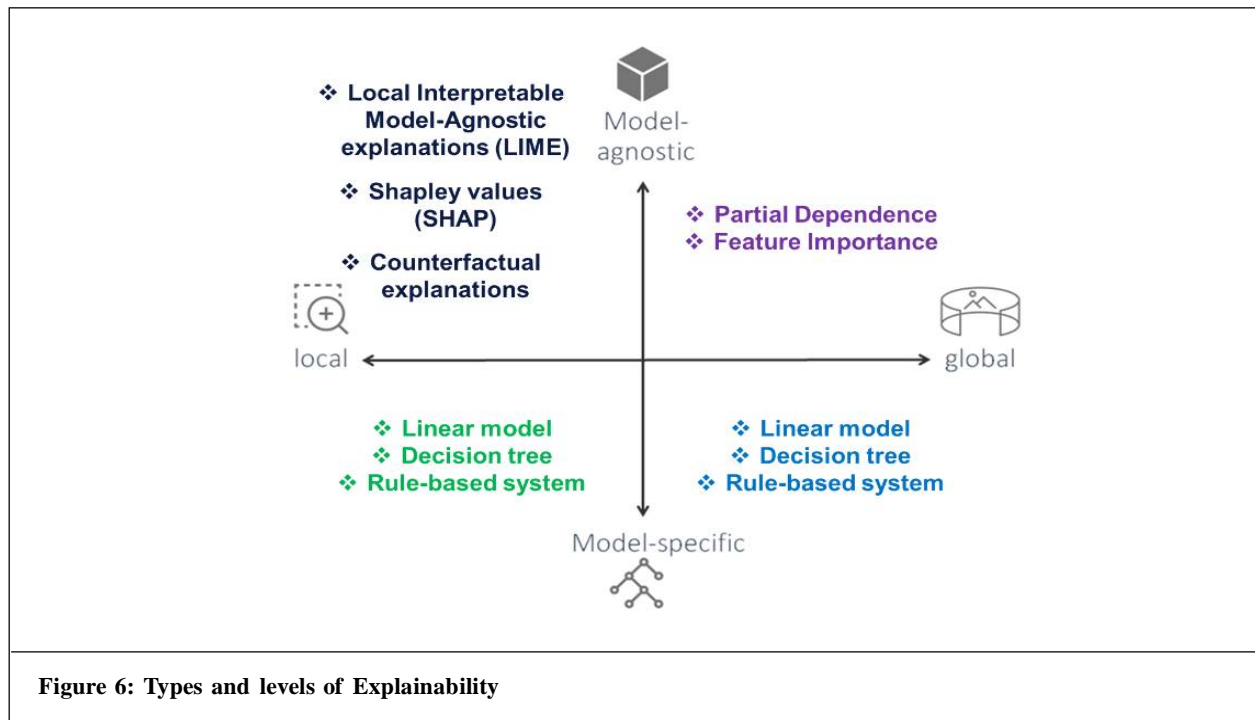
To safely adopt these disrupted techniques in a regulated environment, insurers need to focus on algorithm interpretability/explainability techniques which we will discuss in the next section.

### 5. Interpretability/Explainability of AI algorithms

In the context of AI and ML, Explainability and Interpretability are often used interchangeably. Algorithmic *interpretability* is about the extent to which a cause and effect can be observed within a system, and the extent an observer is able to

predict what will happen, for a given set of input or algorithm parameters. Algorithmic *explainability* is the extent to which the internal mechanics of a ML (deep learning) system is explainable in human terms. In simple terms, Interpretability is about understanding the algorithm mechanics (without necessarily knowing why); explainability is being able to explain what is happening in the algorithm.

There are multiple forms to generate and provide explanations based on an algorithmic decision-making system. Figure 6 presents the types and levels of explainability: model-specific and agnostic, global and local (Molnar, 2020). Below we unwrap these concepts, as well as outline some technical solutions:



**Model-specific:** With model specific explainability, a model is designed and developed in such a way that it is fully transparent and explainable by design. In other words, an additional explainability technique is not required to be overlaid on the model in order to be able to fully explain its workings and outputs. In general, explainable models are simpler than non-explainable models and as such their performance in terms of accuracy is relatively diminished. Explainable models include linear regression, decision trees, k-nearest neighbors, and rule-based systems.

**Model-agnostic:** With model-agnostic explainability, a mathematical technique is applied to the outputs of any algorithm including very complex and opaque models, in order to provide an interpretation of the decision drivers for those models. A few of the most popular approaches include Shapley Explanations (SHAP) (Lundberg and Su-In, 2017) and Local Interpretable Model-Agnostic Interpretation (LIME) (Ribeiro et al., 2016). However, a general limitation of all model-agnostic explainability techniques is that it entails running an additional model on top of an already complex model. The explainability technique will never be 100% accurate, and therefore a layer of additional inaccuracy is introduced into an already inaccurate model, and the output becomes one step further removed from the reality.

**Global:** This facet focuses on understanding the algorithm's behavior at a high/dataset/population level. The usual techniques to provide these explanations are Feature Importance and Partial Dependence. Overall, these methods quantify the weight of a feature in the model's performance and predictions, usually by experimenting with small changes in the data. Apart from a few models (like Decision Trees), both techniques are computationally expensive; it will take time to vary each feature in order to approximate an accurate interpretation of the model, particularly with big datasets. The typical user of Feature Importance and Partial Dependence are researchers and designer of algorithms, since they tend to be more interested with the general insights and knowledge discovery that the model produces, rather than specific individual cases.

**Local:** This facet focuses on understanding the algorithm's behavior at a low/subset/individual level. A variety of methods have been developed in order to help to interpret as to why a model decided for a particular data point. Three of the most popular tools are: LIME; SHAP; Counterfactual explanations (CE) (Wachter et al., 2018). In a nutshell

(i) LIME: samples individual data points and weighs them according to similarity to the individual data point that is to be explained; (ii) SHAP: trains a model with each individual feature, computes the result, repeats with the other features, and then adds features one by one into the model in order to identify the true importance of each feature—this is usually approximated via Monte Carlo sampling; and (iii) CE: this is a computationally expensive technique which considers how the model would behave if some features had different values, allowing an explanation to be built up of individual decision factors, and enabling potential recourse and a more clear understanding by the individual under analysis. The typical user of local explanations are individuals being targeted by an algorithm, as well as members of the judiciary and regulators trying to make a case about potential discrimination.

For other methods and models for Explainable AI we suggest the following reading (Table 5):

| <b>Table 5: Explainable AI methods and models</b>                |   |                          |
|--|---|--------------------------|
| <b>Method</b>  | <b>Description</b>  | <b>Reference</b>         |
| Black Box Explanations through Transparent Approximations (BETA) | Model agnostic framework for explaining the behavior of any black-box classifier by simultaneously optimizing for fidelity to the original model and interpretability of the explanation.   | (Lakkaraju et al., 2017) |
| Layer-wise Relevance Propagation (LRP)                           | Proposed methodology that allows to visualizing the contributions of single pixels to predictions for kernel-based classifiers over Bag of Words features and for multi-layered neural networks.  | (Bach et al., 2015)      |
| Deep Taylor Decomposition  | The iterative application of Taylor decomposition from the top layer down to the input layer.   | (Montavon et al., 2017)  |
| Prediction Difference Analysis                                   | In classification problem, the method highlights areas in a given input image that provide evidence for or against a certain class.   | (Zintgraf et al., 2017)  |
| Deep Generator Networks  | The algorithm: generates qualitatively state-of-the-art synthetic images that look almost real; reveals the features learned by each neuron in an interpretable way,  | (Nguyen et al., 2016)    |
| Testing with Concept Activation Vectors (TCAV)                   | Describes how CAVs may be used to explore hypotheses and generate insights for a standard image classification network as well as a medical application.  | (Kim et al., 2018)       |
| Deep Visualization Recursive Neural                              | Proposes a tool that visualizes the activations produced on each layer of a trained convolutional neural network as it processes an image or video (e.g. a live webcam stream). The second proposed tool enables visualizing features at each layer of a DNN via regularized optimization in image space. | (Yosinski et al., 2015)  |
| Networks cell state analysis                                     | Analysis of their representations, predictions and error types of Recurrent Neural Networks (RNNs), and specifically a variant with Long Short-Term Memory (LSTM).  | (Karpathy et al., 2015)  |

In the next section we will discuss challenges of AI algorithms such as algorithm selection, bias, fairness and robustness of algorithms.

## 6. Governance of algorithms

Here we use the term *governance of algorithms* as a ‘catch all’ for the rules, practices and processes by which an institution directs and controls algorithms and data. This is increasingly important since ML algorithms effectively self-program and evolve dynamically. Hence, financial institutions and regulators are becoming increasingly concerned with issues of Algorithmic *Interpretability/Explainability* and Data *governance*.

### 6.1. Algorithm selection

Users face a number of challenges when selecting algorithms for their application:

- **Backtest overfitting** – where many variations of a (trading) strategy are tried on the same dataset, and, as a result, strategies looking good on paper often perform poorly when presented with new data. Currently, there is an increasing quest for devising a set of procedures to deal with this issue; refer to (Koshiyama and Firoozye, 2019) for a review of the current literature and a few solutions to backtest overfitting.
- **Feature engineering** – is augmentation of data; the process of going from raw data to data that is ready for modeling. Strategies and associated algorithms include: (a) reduce data redundancy/dimensionality (e.g., PCA); (b) capturing complex relationships (e.g., NNs); and (c) rescaling variables (e.g., standardizing or normalizing), etc.
- **Data scarcity** – means too few data points (to train a model) often because it is difficult to get data or the data is small with respect to the amount needed. Whereas *Data sparsity* means data distributed sparsely over the available feature space.
- **Data sensitivity** – data owners need to contribute data to collaborative analytics, while not wishing to ‘share’ extremely valuable and sensitive raw data. An important solution discussed below is Federated Learning.
- **Hyperparameter optimization** – is the problem of choosing a set of optimal input variables (i.e., hyperparameters) for a learning algorithm. A hyperparameter is a parameter whose value is used to control the learning process, in contrast to other parameters (typically node weights) that are ‘learnt’.
- **Interpretability/Explainability** – in machine learning explainability and interpretability are often used interchangeably, but: (a) *Interpretability* is about the extent to which a cause and effect can be observed within a system; and (b) *Explainability* is the extent to which the internal mechanics of an algorithm can be explained in human terms.

Examples of potential solutions to model selection challenges are:

- **Pre-trained models** – a model created to solve a similar problem, often on a large data set, is used as a starting point instead of building a model from scratch. This is the basis of Transfer learning.
- **AutoML** – automated machine learning (AutoML) is the process of automating end-to-end the process of applying machine learning to real-world problems. Companies such as h2o.ai, Datarobot, Amazon, etc. have created *AutoML*-like systems.

### 6.2. Bias and fairness

AI algorithms introduce a significant challenge when it comes to identify where and how algorithms may introduce bias into decision making process. Possible sources of bias:

- **Training set** – skewed sample that, e.g., underrepresents members of protected classes.
- **Proxies** – especially where proxies are hidden within other factors used in machine learning models.
- **Data completeness** – for minority groups’ data might be less informative or reliable.

Unfairness can be mitigated at different points in a modeling pipeline: pre-processing, in-processing and post-processing. Table 6 presents different methodologies to mitigate bias in AI systems:

| <b>Pipeline</b> | <b>Technical solution</b>  |
|-----------------|--|
| Pre-processing  | <ul style="list-style-type: none"> <li>• Reweighting subjects</li> <li>• Oversampling minority group</li> <li>• Disparate impact remover</li> <li>• Learning fair representations</li> </ul> |
| In-processing   | <ul style="list-style-type: none"> <li>• Adversarial debiasing</li> <li>• Fairness constraint</li> <li>• Counterfactual fairness</li> </ul>  |
| Post-processing | <ul style="list-style-type: none"> <li>• Calibrated equality of odds</li> <li>• Reject option classification</li> </ul>  |



### 6.3. Robustness

Algorithmic robustness is characterized by how effectively an algorithm can be deemed as safe and secure, not vulnerable to tampering or compromising of the data they are trained on. We can rate an algorithm's robustness using four key criteria (EU-HLEG 2019):

- **Resilience to attack and security:** AI systems, like all software systems, should be protected against vulnerabilities that can allow them to be exploited by adversaries, such as data poisoning, model leakage or the infrastructure, both software and hardware. This concept is linked with the mathematical concept of *Adversarial Robustness* (Carlini et al., 2019), that is, how would the algorithm performed in the worst-case scenario? (e.g., how the algorithm would react during the 2008 Financial Crisis?).
- **Fallback plan and general safety:** AI systems should have safeguards that enable a fallback plan in case of problems. Also, the level of safety measures required depends on the magnitude of the risk posed by an AI system. This notion is strongly associated with the technical concept of *Formal Verification* (Qin et al., 2019), which in broad terms means: does the algorithm attends the problem specifications and constraints? (e.g., respect physical laws).
- **Accuracy:** pertains to an AI system's ability to make correct judgments, for example to correctly classify information into the proper categories, or its ability to make correct predictions, recommendations, or decisions based on data or models. Accuracy as a general concept can be quantified by estimating the *Expected Generalization Performance* (Arlot and Calisse, 2009), which means that in general, how well the algorithm works? (e.g., in 7 out of 10 cases, the algorithm makes the right decision).
- **Reliability and reproducibility:** a reliable AI system is one that works properly with a range of inputs and in a range of situations, whilst reproducibility describes whether an AI experiment exhibits the same behavior when repeated under the same conditions. This idea is tied with the software engineering concept of *Continuous Integration* (Meyer, 2014) TI, that is, is the algorithm auditable? (e.g., reliably reproduce its decisions).

In practice, each technical criteria embodies a number of technical solutions (Table 7). These technical solutions can aid the analyst in measuring and having systems in place to assess and make systems more robust before deployment stage.

| Table 7: Mapping technical criteria and solutions for algorithmic robustness |   |
|--|---|
| Criteria   | Technical solution  |
| Expected generalization performance  | <ul style="list-style-type: none"> <li>• Cross-validation: k-fold cv, bootstrap, etc.</li> <li>• Covariance-penalty: Mallows's <math>C_p</math>, Stein Unbiased Risk Estimator, bootstrap approximation, etc.</li> </ul>            |
| Adversarial robustness   | <ul style="list-style-type: none"> <li>• Evasion attacks: fast gradient sign method, DeepFool, etc.</li> <li>• Defence: label smoothing, variance minimization, etc.</li> </ul>   |
| Formal verification  | <ul style="list-style-type: none"> <li>• Complete: Satisfiability Modulo Theory, Mixed Prog., etc.</li> <li>• Incomplete: Propagating bounds, Convex Optimization, etc.</li> </ul>  |
| Reliability and reproducibility  | <ul style="list-style-type: none"> <li>• Code versioning: Git (Github), Mercurial (BitBucket), etc.</li> <li>• Reproducible analysis: Binder, Docker, etc.</li> <li>• Automated testing: Travis CI, Scrutinizer CI, etc.</li> </ul> |

### 6.4. Risk

Algorithms, especially so-called black box trading algorithms, amplifies systemic risk for a number of reasons: (a) *Intensifying Volatility* – algorithms can react instantaneously to market conditions and during volatile markets may greatly widen their bid-ask spreads, or temporarily stop trading thereby diminishing liquidity; (b) *Flash crash* – increased algorithm and market integration means a meltdown in a major market or asset class often has a ripple effect across other markets; (c) *Uncertainty* – algorithm opaqueness stokes investor uncertainty; (d) *Rogue algorithms* – due to speed and lack of transparency one errant or faulty algorithm can rack up millions in losses in a very short period (e.g., Knight

Capital lost \$440 mn in a 45 min period on August 1, 2012); and (e) *Algorithm uniformity* – a lack of diversity in (trading) algorithms could reduce robustness in a market (cf. Irish potato famine).

### 6.5. Legality and ethics

Increasingly, ML algorithms self-program and evolve dynamically, raising concerns about explainability of financial decisions (e.g., for mortgages, loans); discriminatory, unethical and illegal behavior (e.g., CV/Resume ‘sifting’ recruitment systems); and unintentional fraudulent systems (e.g., systematic trading systems and market manipulation). Naturally, financial institutions and regulators are becoming increasingly concerned with ensuring there remains a modicum of human control.

### 6.6. Compliance and regulation

Compliance departments are increasingly using AI algorithms to automate procedures and monitor behavior of employees. Now they face the increasing challenge of self-programming algorithms being discriminatory, unethical and illegal; exposing the institutions to (unintentional) reputational damage, financial loss and potentially large fines.

Financial regulators are also increasing using AI algorithms to automate monitoring and reporting (Treleaven et al., 2019). To underpin this automation and leverage AI algorithms, leading financial regulators are seeking to encode regulatory rules as computer-executable code, allowing compliance and regulation to be fully automated, and operate in real time and across multiple jurisdictions.

Traditionally, regulators have faced the challenge of regulating institutions, individuals, and processing the ‘tsunami’ of reporting data. Going forward, regulators have the additional challenge of regulating algorithm behavior.

### 6.7. Legal status of algorithms

Finally, there is the growing discussion in the Judiciary concerning the ‘status of algorithms in Law’. In Law, as we know, companies have the rights and obligations of a person. Algorithms are rapidly emerging as artificial persons: a legal entity that is not a human being but for certain purposes is legally considered to be a natural person. The argument is that since algorithms are doing or intermediating business (agency) with humans, companies and even other algorithms they also need to have the status of an artificial person in Law.

### 6.8. Alternative data

However, although AI and algorithms receive all the publicity, many people believe that we have yet to experience the full extent of the so-called data revolution, and especially the use by the capital markets of alternative data. For example, investment funds are buying anonymized real-time credit card data, and therefore can ‘see’ what is going through a retailer’s tills and expansively across the whole industry sector. As a definition, alternative data (in finance) refers to data used to obtain insight into the investment process; sources such as financial transactions, retail data, sensors, mobile devices, satellites, public records, and the Internet. Surprisingly companies that produce alternative data (e.g., credit card, retail, telecoms, transportation, etc.) generally overlook the value of their data to financial institutions. Hence, these data sets are often less readily accessible and less well-structured than traditional sources of data. Refer to the (Denev and Amen, 2020) book for a comprehensive introduction and review in this topic.

## 7. Conclusion

This paper reviews AI, ML and associated algorithms, and discusses their future impact on the Insurance Markets. The data science technologies driving change include: Big data, AI analytics, Internet of Things, and Blockchain technologies. These technologies are important since they underpin the automation of the Insurance Markets and risk analysis, and provide the context for the algorithms, such as AI machine learning and computational statistics, which provide powerful analytics capabilities.

The current main disrupting forms of learning include Deep Learning, Adversarial Learning, Federated Learning, Transfer and Meta Learning. These forms of learning have produced new models (e.g., LSTM, GANs) and leverage important applications (e.g., Natural Language Processing, Adversarial Examples, Deep Fakes, etc.) which we have discussed more in this review. Risk management, marketing, reserving, claims handling are only the examples of areas where AI will empower insurance.

Companies also need to embrace automation with digital infrastructure:

**Data Management Solutions** – Successful Insurance companies will collect increasing amounts of historic and real-time data (e.g., business, economic, social media, alternate) to drive business decisions.

**Automated trading Platform** – Digital marketplaces and platforms provide direct engagement with clients, support automation and drive down costs.

Undoubtedly, data revolution will be fuelled by the upgrade of technology frameworks used across the Industry (e.g., Agile development, data lakes, microservices, Python, Cloud computing, Hadoop, NoSQL databases).

## References

- Adewumi, A.O. and Akinyelu, A.A., (2017). [A survey of machine-learning and nature-inspired based credit card fraud detection techniques. \*International Journal of System Assurance Engineering and Management\*, 8, 937-953.](#)
- Allan, N., Cante, N., Godfrey, P. and Yin, Y. (2012). [A review of the use of complex systems applied to risk appetite and emerging risks in ERM practice: Recommendations for practical tools to help risk professionals tackle the problems of risk appetite and emerging risk.. \*British Actuarial Journal\*, 18\(1\), 163-234.](#)
- Arlot, S. and Calisse, A. (2009). [A survey of cross-validation procedures for model selection. \*arXiv:0907.4728\*.](#)
- Arulkumaran, K., Deisenroth, M.P., Brundage, M. and Bharath, A.A. (2017). [Deep reinforcement learning: a brief survey. \*IEEE\*, 34\(6\), 26-38.](#)
- Bach, S. et al. (2015). [On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. \*PLOS ONE\*.](#)
- Batrinca, B. and Treleaven, P.C. (2015). [Social media analytics: a survey of techniques, tools and platforms. \*AI and Society\*, 30, 89-116.](#)
- Bäumer, H. P. (2000). [Statistical analysis of extreme values - from insurance, finance, hydrology and other fields. \*Computational Statistics\*, 5, 307-311.](#)
- Brock, A., Donahue, J. and Simonyan, K. (2018). [Large scale gan training for high fidelity natural image synthesis. \*arXiv:1809.11096\*.](#)
- Brownlee, J. (2011). [Clever Algorithms: Nature-Inspired Programming Recipes. s.l.:Lulu.com](#)
- Cambria, E., Schuller, B. and Liu, B. (2013). [Statistical approaches to concept-level sentiment analysis. \*Intelligent Systems, IEEE\*, 28\(3\), 6-9.](#)
- Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D., Goodfellow, I., Madry, A., and Kurakin, A. (2019). [On evaluating adversarial robustness. \*arXiv:1902.06705\*](#)
- Caron, M., Bojanowski, P., Joulin, A. and Douze, M. (2018). [Deep clustering for unsupervised learning of visual features.. \*The European Conference on Computer Vision \(ECCV\)\*, pp. 132-149.](#)
- Chen, S., Ma, K. and Zheng, Y. (2019). [Transfer learning for 3d medical image analysis. \*arXiv preprint arXiv:1904.00625\*](#)
- Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I. and Abbeel, P. (2016). [InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets.. \*arXiv:1606.03657\*.](#)
- Chollet, F. (2017). [Deep Learning with Python. s.l.:Manning.](#)
- Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Biswa Sengupta, B. and Bharath, A.A. (2018). [Generative adversarial networks: An overview. \*IEEE Signal Processing Magazine\*, 35\(1\), 53-65.](#)
- Dai, W., Xue, G. R., Yang, Q. and Yu, Y. (2007). [Co-clustering based classification for out-of-domain documents. \*KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining\*, pp. 210-219.](#)
- Denev, A. and Amen, S. (2020). [The Book of Alternative Data: A Guide for Investors, Traders and Risk. s.l.:Wiley.](#)
- Devlin, J., Chang, M.W., Lee, K. and Toutanova, K. (2018). [BERT: Pre-training of deep bidirectional transformers for language understanding. \*arXiv:1810.04805\*.](#)
- Diao, L. and Wang, N. (2019). [Research on premium income prediction based on LSTM neural network. \*Advances in Social Sciences Research Journal\*, 6\(11\), 256-260.](#)
- Díaz, Z., Segovia, M.J., Fernández, J. and del Pozo, E.M. (2005). [Machine learning and statistical techniques. An application to the prediction of insolvency in spanish non-life insurance companies.. \*The International Journal of Digital Accounting Research\*, 5\(9\), 1-45.](#)
- Efron, B. and Hastie, T. (2016). [Computer Age Statistical Inference: Algorithms, Evidence and Data Science. Cambridge University Press, 5.](#)

- Fei-Fei, L., Fergus, R. and Perona, P. (2003). A bayesian approach to unsupervised one-shot learning of object categories. *In Proceedings Ninth IEEE International Conference on Computer Vision*, 1134-1141.
- Giarratano, J.C. and Riley, G. (1998). *Expert Systems*. s.l.:PWS Publishig Co..
- Gleave, A., Dennis, M., Wild, C., Kant, N., Levine, S. and Russell, S. (2019). Adversarial policies: attacking deep reinforcement learning.. *arXiv:1905.10615*.
- Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*. s.l.:MIT Press book.
- Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014). Generative Adversarial Networks. *arXiv:1406.2661*.
- Graves, A. (2014). Generating sequences with recurrent neural networks. *arXiv:1308.0850*.
- Graves, A. and Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. *Proceedings of the 31st International Conference on Machine Learning*, 32(2), 1764-1772.
- Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H. and Schmidhuber, J. (2009). A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5), 855-868.
- Graves, A., Mohamed, A.R. and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. *arXiv:1303.5778*.
- Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Second ed. s.l.:Springer.
- Higgins, I., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H. and Schmidhuber, J. (2017). DARLA: Improving zero-shot transfer in reinforcement learning. *arXiv:1707.08475*.
- Hochreiter, S. and Schmidhuber, J. (2006). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.
- Huang, L., Joseph, A.D., Nelson, B., Rubinstein, B.I.P. and Tygar, J.D. (2011). Adversarial machine learning. *AISeC '11: Proceedings of the 4th ACM workshop on Security and artificial intelligence*, pp. 43-58.
- Ilker, K., Mehmet, G. and Kemal, K. (2015). An interactive machine-learning-based electronic fraud and abuse detection system in healthcare insurance.. *Applied Soft Computing*, 36, 283-299.
- Juszczak, P., Adams, N.M., David J.Hand, D.J., Whitrow, C. and Westona, D.J. (2008). Off-the-peg and bespoke classifiers for fraud detection. *Computational Statistics and Data Analysis*, 52(9), 4521-4532.
- Karpathy, A., Johnson, J. and Fei-Fei, L. (2015). Visualizing and understanding recurrent networks. *arXiv:1506.02078v2*.
- Keneshloo, Y., Ramakrishnan, N. and Reddy, C. K. (2019). Deep transfer reinforcement learning for text summarization. *arXiv:1810.06667*.
- Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F. and Sayres, R. (2018). Interpretability beyond feature attribution: quantitative testing with concept activation vectors (TCAV). *Proceedings of the 35th International Conference on Machine Learning*, pp. 2668-2677.
- Kiros, R., Salakhutdinov, R. and Zemel, R.S. (2014). Unifying visual-semantic embeddings with multimodal neural language models. *arXiv:1411.2539*.
- Kolchyna, O., Souza, T.P. and Treleaven, P. (2015). Twitter sentiment analysis. *arXiv:1507.00955*.
- Koshiyama, A. and Firoozye, N. (2019). Avoiding backtesting overfitting by covariance-penalties. *The Journal of Financial Data Science*.
- Koshiyama, A., Firoozye, N. and Treleaven, P. (2020). Algorithms in future capital markets. SSRN: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3527511](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3527511)
- Kuo, K. (2019). Generative synthesis of insurance datasets. *arXiv:1912.02423*.
- Kurakin, A., Goodfellow, I. and Bengio, S. (2016). Adversarial machine learning at scale. *arXiv:1611.01236*.
- Lakkaraju, H., Kamar, E., Cauruna, R. and Leskovec, J. (2017). Interpretable and explorable approximations of black box models. *arXiv:1707.01154*.
- Li, Q. et al. (2019). A survey on federated learning systems: vision, hype and reality for data privacy and protection. *arXiv:1907.09693*.

- Lundberg, S. and Su-In, L. (2017). A unified approach to interpreting model predictions. *arXiv:1705.07874*.
- Makhzani, A. et al. (2016). Adversarial autoencoders. *arXiv:1511.05644*.
- Marano, P. and Noussia, K. (2019). Insurtech: a legal and regulatory view. In: *Smart Contracts in Insurance: A Law and Futurology Perspective*. s.l.:Springer, pp. 101-125.
- May, R.M., Levin, S.A. and Sugihara, G. (2008). Complex systems. ecology for bankers. *Nature*. 451, 893-893.
- Meyer, M. (2014). Continuous integration and its tools. *IEEE Software*, 31(3), 14-16.
- Miyato, T., Dai, A.M. and Goodfellow, I. (2017). Adversarial training methods for semi-supervised text classification.. *arXiv:1605.07725*.
- Molnar, C. (2020). *Interpretable machine learning*. s.l.:Lulu.com.
- Montavon, G., Samek, W. and Müller, K.R. (2017). Methods for Interpreting and Understanding Deep Neural Networks. *arXiv:1706.07979*.
- Nguyen, A. et al. (2016). Synthesizing the preferred inputs for neurons in. *Advances in Neural Information Processing Systems 29 (NIPS 2016)*.
- Nicolae, M.I. et al. (2019). Adversarial robustness toolbox. v0.4.0. *arXiv:1807.01069*.
- Pinto, L., Davidson, J., Sukthankar, R. and Gupta, A. (2017). Proceedings of the 34th International Conference on Machine Learning. Volume 70.
- Poli, R., Langdon, W.B. and McPhee, N.F. (2008). *A field guide to genetic programming*. s.l.:Lulu.com.
- Qin, C. et al. (2019). Verification of non-linear specifications for neural networks. *arXiv:1902.09592*.
- Radford, A. et al. (n.d.). Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).
- Raina, R., Ng, A. Y. and Koller, D. (2006). Constructing informative priors using transfer learning. *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pp. 713-20.
- Ribeiro, M.T., Singh, S. and Guestrin, C. (2016). Why should i trust you?: explaining the predictions of any classifier. *arXiv:1602.04938*.
- Richman, R. (2018). AI in actuarial science. [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3218082](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3218082)
- Russell, S. and Norvig, P. (2020). *Artificial intelligence: a modern approach*. 4th ed. s.l.:Pearson.
- Saleh, K., Hossny, M. and Nahavandi, S. (2017). Driving behavior classification based on sensor data fusion using LSTM recurrent neural networks. *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*.
- Saleh, K., Hossny, M. and Nahavandi, S. (2017). Driving behavior classification based on sensor data fusion using LSTM recurrent neural networks. *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*.
- Siddhant, A., Goyal, A. and Metallinou, A. (2019). Unsupervised transfer learning for spoken language understanding in intelligent agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Volume 4959-4966.
- Sutskever, I., Vinyals, O. and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *arXiv:1409.3215*.
- Sutton, R.S. and Barto, A.G. (2018). *Reinforcement learning: An introduction*. s.l.:MIT Press.
- Szegedy, C. et al. (2013). Intriguing properties of neural networks. *arXiv:1312.6199*.
- Taylor, S. J. (2014). *Agent-Based Modeling and Simulation*. s.l.:Springer.
- Treleaven, P., Barnett, J. and Koshiyama, A. (2019). Algorithms: Law and regulation. *Computer*, 52(2), 32-40.
- Treleaven, P., Galas, M. and Lalchand, V. (2013). Algorithmic trading review. *Communications of the ACM*, 56(11), 76-85.
- van Liebergen, B. (2017). Machine learning: A revolution in risk management and compliance?. *Journal of Financial Transformation*, 45, 60-67.
- Vinyals, O. et al. (2015). Grammar as a foreign language. *arXiv:1412.7449*.
- Wachter, S., Mittelstadt, B. and Russell, C. (2018). Counterfactual explanations without opening the black box: Automated Decisions and the GDPR. *Harvard Journal of Law and Technology*, 31(2).
- Weishan, D. et al. (2016). Characterizing driving styles with deep learning. *arXiv:1607.03611*.



- Wood, S. N. (2017). *Generalized additive models*. s.l.:Chapman and Hall/CRC.
- Wüthrich, M. V. (2018). Machine learning in individual claims reserving.. *Scandinavian Actuarial Journal*, 465-480.
- Xu, J. and Chen, H. (2005). Criminal network analysis and visualization. *Communications of the ACM*, 48(6), 100-107.
- Xu, K. et al. (2016). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *arXiv:1502.03044*
- Yang, Q. et al. (2019). *Federated learning. synthesis lectures on artificial intelligence and machine learning*. s.l.:Morgan and Claypool Publishers.
- Yang, W. and Koshiyama, A.S. (2019). Assessing qualitative similarities between financial reporting frameworks using visualization and rules: COREP vs. pillar 3.. *Intelligent Systems in Accounting, Finance and Management*, 26(1), 16-31.
- Yaqi, L., Chun, Y., Wei, L. and Maozhen, L. (2018). principle component analysis-based random forest with the potential nearest neighbor method for automobile insurance fraud identification. *Applied Soft Computing*, 70(September 2018), 1000-1009.
- Yosinski, J. et al. (2015). Understanding neural networks through deep visualization. *arXiv:1506.06579*.
- Zintgraf, L.M., Chohen, T.S., Adel, T. and Welling, M. (2017). Visualizing deep neural network decisions: prediction difference analysis. *arXiv:1702.04595*.